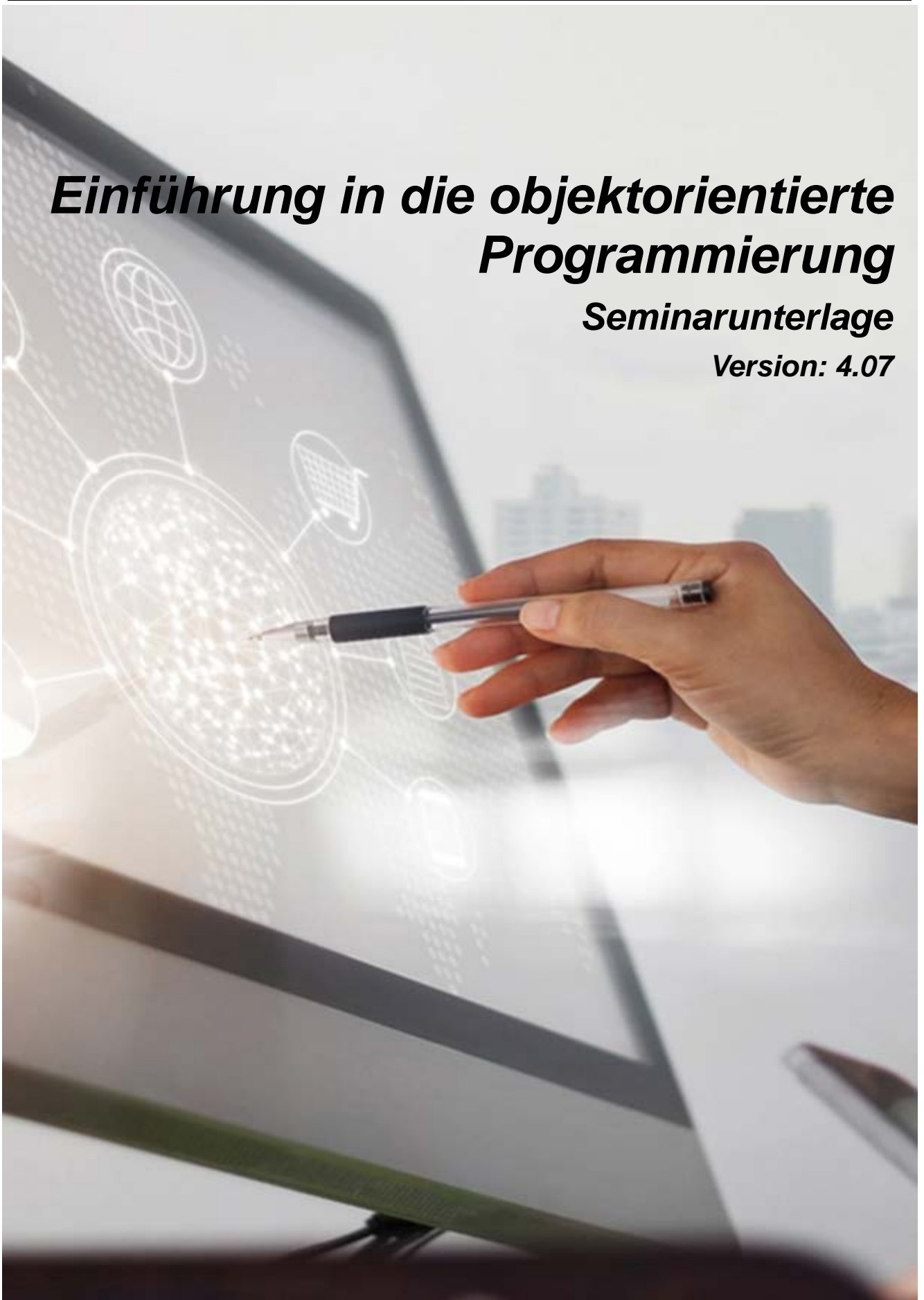


Einführung in die objektorientierte Programmierung

Seminarunterlage

Version: 4.07



Dieses Dokument wird durch die ORDIX AG veröffentlicht.

Copyright ORDIX AG. Alle Rechte vorbehalten.

Alle Produkt- und Dienstleistungs-Bezeichnungen sind Warenzeichen oder eingetragene Warenzeichen der jeweiligen Firmen und beziehen sich auf Eintragungen in den USA oder USA-Warenzeichen.

Weitere Logos und Produkt- oder Handelsnamen sind eingetragene Warenzeichen oder Warenzeichen der jeweiligen Unternehmen.

Kein Teil dieser Dokumentation darf ohne vorherige schriftliche Genehmigung der ORDIX AG weitergegeben oder benutzt werden.

Adressen der ORDIX AG

Die ORDIX AG besitzt folgende Geschäftsstellen

ORDIX AG
Karl-Schurz-Straße 19a
D-33100 Paderborn
Tel.: (+49) 0 52 51 / 10 63 - 0
Fax.: (+49) 01 80 / 1 67 34 90

ORDIX AG
An der alten Ziegelei 5
D-48157 Münster
Tel.: (+49) 02 51 / 9 24 35 – 00
Fax.: (+49) 01 80 / 1 67 34 90

ORDIX AG
Welser Straße 9
D-86368 Gersthofen
Tel.: (+49) 08 21 / 507 492 – 0
Fax.: (+49) 01 80 / 1 67 34 90

ORDIX AG
Kreuzberger Ring 13
D-65205 Wiesbaden
Tel.: (+49) 06 11 / 7 78 40 – 00
Fax.: (+49) 01 80 / 1 67 34 90

ORDIX AG
Wikingerstraße 18-20
D-51107 Köln
Tel.: (+49) 02 21 / 8 70 61 – 0
Fax.: (+49) 01 80 / 1 67 34 90

ORDIX AG
Südwestpark 67/2
D-90449 Nürnberg
Tel.: (+49) 0 52 51 / 10 63 - 0
Fax.: (+49) 01 80 / 1 67 34 90

Internet: <http://www.ordix.de>

Email: seminare@ordix.de

Inhaltsverzeichnis

1	Motivation.....	6
1.1	Motivation.....	7
2	Grundlagen der Objektorientierung	8
2.1	Überblick	9
2.2	Wieso objektorientierte Programmentwicklung.....	10
2.3	Abstraktion	12
2.4	Objektorientierter oder prozeduraler Ansatz?	13
2.5	Objekte	16
2.5.1	Zustand und Verhalten	17
2.5.2	Kapselung und Information-Hiding.....	18
2.5.3	Schaubild Objekt	20
2.5.4	Objekt-Kommunikation	21
2.6	Klassen.....	22
2.7	Objekte und Klassen	24
2.8	Objekte und Klassen in UML	25
2.9	Klasseneigenschaften	27
2.9.1	Attribute	28
2.9.2	Operationen.....	29
2.9.3	Zusicherungen.....	30
2.10	Klasseneigenschaften in UML	31
2.11	Beziehungen zwischen Klassen: Assoziation	33
2.12	Beziehungen zwischen Klassen: Aggregation	34
2.13	Beziehungen zwischen Klassen: Komposition.....	35
2.14	Nachrichtenaustausch.....	37
2.15	Nachrichtenaustausch in UML	39
2.15.1	Nachrichtenaustausch: Sequenzdiagramm	40
2.15.2	Nachrichtenaustausch: Kommunikationsdiagramm.....	41
2.16	Basiselemente der Objektorientierung.....	42
2.16.1	Datenkapselung	43
2.16.2	Vererbung.....	45
2.16.2.1	Klassen – Vererbung	45
2.16.2.2	Taxonomie und Vererbung.....	46
2.16.2.3	Mehrfachvererbung	48
2.16.2.4	Vorteile der Vererbung	49
2.16.2.5	Vererbung in UML	50
2.16.2.6	Klassendiagramm Vererbung.....	51
2.16.2.7	Abstrakte Klassen in UML	53
2.16.3	Polymorphie	54
2.16.3.1	Statische Polymorphie.....	55
2.16.3.2	Dynamische Polymorphie.....	56
2.16.3.3	Beispiel der Dynamischen Polymorphie.....	57
2.17	Entwurfsmuster	58
2.17.1	Beispiel: Model-View-Controller	59
2.17.2	Beispiel: Kompositum (Composite).....	60
2.18	Persistenz.....	62
2.19	Vorteile objektorientierter Softwareentwicklung.....	63
3	Objektorientierte Analyse und Design	64
3.1	Ziel der Analyse	65
3.2	Anforderungsanalyse	66
3.2.1	Anwendungsfallanalyse.....	67
3.2.1.1	Beispiel einer Anwendungsfallanalyse	69
3.3	Anwendungsfälle beschreiben	71
3.4	Eigenschaften von Anwendungsfällen	72
3.5	Systemvoraussetzungen.....	73
3.6	Anwendungsarchitektur.....	74
3.6.1	Komponenten einer Anwendungsarchitektur	76

3.7	Fachlexikon	77
3.8	Prototyp der Benutzeroberfläche	78
3.9	Explorative Prototypen	79
3.10	Dialog-Workshop	81
3.11	Geschäftsklassen identifizieren	82
3.12	Aktivitätsmodellierung	84
3.12.1	Aktivitätsmodellierung mit Objektzuständen	86
3.13	Komponentenbildung	87
3.13.1	Komponentendiagramm	89
3.14	OO-Design	90
3.14.1	Abgrenzung von Analyse und Design	90
3.14.2	Übergang OOA - OOD – OOP	91
3.15	Design	92
3.15.1	Komponentenentwurf	94
3.15.2	Dialoge spezifizieren	95
3.15.3	Dialogkomponenten	96
3.16	Fachklassen und Beziehungen identifizieren.....	98
3.16.1	Beispiel.....	99
3.17	Operationen spezifizieren	110
3.18	Attribute spezifizieren	111
3.19	Aktivitäten modellieren	112
3.20	Zustände modellieren.....	114
3.21	Objektinteraktionen modellieren	116
3.22	Hinweise.....	118
3.22.1	Designregeln und -heuristiken	118
3.22.2	Namenskonventionen.....	119
3.23	OOD Modelle	120
3.24	OO-Programmierung.....	121
4	Unified Modeling Language	122
4.1	Allgemeines.....	123
4.2	Geschichte / Gliederung der UML.....	124
4.3	Struktur- und Verhaltensdiagramm	125
4.4	Klassendiagramm	126
4.5	Klasse.....	127
4.6	Abstrakte Klasse	128
4.7	Objekt.....	129
4.8	Attribut.....	130
4.9	Operation, Methode	131
4.10	Schnittstelle, Schnittstellenklasse	133
4.11	Entwurfsmuster	135
4.12	Paket	136
4.13	Beispiel: Paketdiagramm	137
4.14	Klassendiagramm	138
4.15	Generalisierung, Spezialisierung	139
4.16	Assoziation	141
4.17	Aggregation	143
4.18	Komposition	144
4.19	Abhängigkeitsbeziehung.....	145
4.20	Verfeinerungs- und Realisierungsbeziehung.....	146
4.21	Anwendungsfalldiagramm.....	147
4.22	Anwendungsfall.....	148
4.23	Akteur	149
4.24	Anwendungsfalldiagramm.....	150
4.25	Verhaltensdiagramme	152
4.26	Aktivitätsdiagramm	153
4.27	Sequenzdiagramm	155
4.28	Beispiel: Sequenzdiagramm	156
4.29	Sequenzdiagramm: Erweiterungen.....	157
4.30	Zustandsdiagramm	158

4.31	Zustand	159
4.32	Zustand: Notation	160
4.33	Beispiel: Zustand.....	161
4.34	Ereignis und Zustandsübergang	162
4.35	Implementierungsdiagramme	163
4.36	Komponentendiagramm.....	164
4.37	Verteilungsdiagramm	165
4.38	Beispiel: Verteilungsdiagramm	166
5	Entwurfsmuster	167
5.1	Definition	168
5.2	Elemente eines Entwurfsmusters	169
5.3	Kategorien	170
5.4	Singleton	171
5.5	Umsetzung in Java.....	172
5.6	Composite	173
5.7	Strategy	175
5.8	Proxy	177
5.9	Observer.....	179